TEC-0079

Research in Model-Based Change Detection and Site Model Updating

R. Nevatia

University of Southern California
Institute for Robotics & Intelligent Systems
Powell Hall 204
Los Angeles, CA 90089-0273

January 1996

Prepared for:
Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

Monitored by:
U.S. Army Corps of Engineers
Topographic Engineering Center
7701 Telegraph Road
Alexandria, Virginia 22315-3864

US Army Corps of Engineers
Topographic Engineering Center

T

E

C

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 1996	First Annual July 1993 - July 1994

4. TITLE AND SUBTITLE

Research in Model-Based Change Detection and Site Model Updating

5. FUNDING NUMBERS

DACA76-93-C-0014

6. AUTHOR(S)

R. Nevatia

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Southern California
Institute for Robotics & Intelligent Systems
Powell Hall 204
Los Angeles, CA 90089-0273

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Advanced Research Projects Agency
3701 North Fairfax Drive, Arlington, VA 22203-1711

U.S. Army Topographic Engineering Center
7701 Telegraph Road, Alexandria, VA 22315-3864

19. SPONSORING / MONITORING AGENCY REPORT NUMBER

TEC-0079

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Change detection is perhaps the most important task in the process of photo-interpretation, and an ideal one to demonstrate the effectiveness of the site modeling exploitation (SME) approach that has been adopted for the RADIUS program. Change detection is a tedious task as it requires careful comparison of images (and their models) taken at different times under possibly varying conditions. The task of change detection consists of comparing a new image of a site (or a colletion of images), to the information contained in the folder for that site. The information in the site folder may consist of one or more previous images and results of previous analyses on these images. We believe that even partial automation of this task will greatly increase an analyst's productivity and possibly enhance the reliability of the results.

14. SUBJECT TERMS

Change Detection, Fine Registration, Matching, Perceptual Grouping, Shadow Analysis, Model Construction

15. NUMBER OF PAGES

44

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

TABLE OF CONTENTS

ILLUSTRATIONS

FIGURE **PAGE**

TABLES

TABLE **PAGE**

Preface

This research is sponsored by the Advanced Research Projects Agency (ARPA) and monitored by the U.S. Army Topographic Engineering Center (TEC) under contract DACA76-93-C-0014, titled "Research in Model-Based Change Detection and Model Updating." The ARPA Prohram Manager is Dr. Oscar Firschein, and the TEC Contracting Officer's Representative is Ms. Lauretta Williams.

1 Introduction

We have undertaken research on model-based change detection and site model updating. Change detection is perhaps the most important task in the process of photo-interpretation and an ideal one to demonstrate the effectiveness of the site modeling exploitation (SME) approach that has been adopted for the (Research And Development for Image Understanding Systems (RADIUS) program. Change detection is a tedious task as it requires careful comparison of images (and their models) taken at different times under possibly vastly varying conditions. We believe that even partial automation of this task will greatly increase an analyst's productivity and possibly also enhance the reliability of the results. Furthermore, change detection offers a challenging (Image Understanding) IU research opportunity for which some of the foundation has been laid.

The task of change detection consists of comparing a new image of a site (or a collection of images) to the information contained in the *folder* for that site. The information in the site folder may consist of one or more previous images and results of previous analyses on these images. We assume that in all cases, a site model of suitable resolution and complexity is included in the site folder (though we may still need to examine the older images and other data directly). Also, collateral information about the site may be available.

The task of change detection consists of finding significant differences between the new data and the model derived from the older data. The significance of the differences may be task specific, though in most cases, man-made changes are more important than changes caused by natural factors, such as seasonal changes. The differences need to be described not so much in terms of the changes in the image, but in changes in the site. Functional inferences need to be drawn from the detected changes as well.

Complete automation of the highly complex change detection task would require the implementation of virtually all of the other tasks of RADIUS, including that of site modeling. Clearly we can not solve all these problems in this effort of modest size and will need to select problems carefully. The following describes the rationale for selecting some problems and proposed approaches for solving them. Section 2 discusses progress on model registration and validation, the first stage towards the development of a change detection system. Section 3 discusses the status of automatic model construction techniques applied to building structures. It should be recognized that, to a certain extent, the choice of problems will be influenced by continuing tests with image analysts in the RADIUS program and availability of suitable data.

1.1 Phases in Change Detection

It is useful to break the change detection task into the following sub-tasks that have close IU analogs.

- *Detection*: In this phase, we determine whether a *significant* change has taken place since the last look at the site.
- *Description*: In this phase, a description of the change is obtained. The description may consist of the size and shape of the new (or altered) structure and surface properties. This step is similar to the process of constructing a site model, and, in fact, one result of this step may be an updated site model.
- *Functional Inference*: In this phase, an attempt is made to judge the purpose of the change and the role that the new structure may play in the function of the overall site. This step is like the reasoning processes in Artificial Intelligence (AI), but needs to handle geometrical objects and relations.

1.2 Types of Changes

The kinds of changes that may be of interest can be characterized as follows:

- *Changes to Existing objects*: Significant changes are made to existing structures, for example, a new wing is added to an existing building, a road is widened, or a runway [8] is lengthened.
- *New Objects*: Here new objects appear at the site, such as a new building, new bridge or a new power line.
- *Preparation for Construction*: Here the new structure may not be apparent but preparations for construction are visible. Examples are earth movement for foundation, presence of construction equipment and/or materials, or clearing of forested areas.
- *Changes of detail*: Here small but significant details have changed in existing structures. An example is a new antenna on the roof of an existing building.
- *Redeployment*: Here mobile objects, such as vehicles and aircraft are moved around and redeployed.

1.3 Process of Change Detection

The task of change detection can be broken into two sub-tasks: *Detection* of changes and *Description* of changes. In the detection phase we determine whether a *significant* change has taken place since the last look at the site. In the description phase, a description of the change is obtained. The description may consist of the size and shape of the new (or altered) structure and surface properties. This step is similar to the process of constructing a site model, and, in fact, one result of this step may be an updated site model.

Figure 1.1 shows a flowchart of the complete change detection system. The process requires a comparison of new imagery with the old (and the models constructed

from the old imagery). However, a distinction needs to be made between changes in the images and changes in the site. We are only interested in those changes in the image that come from some changes in the site rather than from changes in imaging conditions. The techniques of simple image differencing are inadequate for this task.

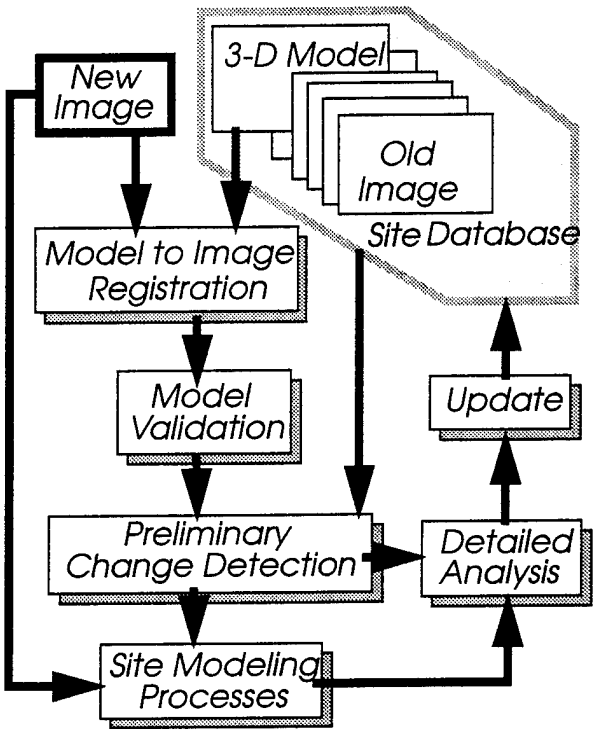


Figure 1.1 Flowchart of change detection system.

We follow a four step approach:

- *Registration of Site Model to Image*: The first step in change detection is to register the new image(s) to the model(s) contained in the site folder. We have assumed that this ability will be available from other ongoing RADIUS projects.
- *Model Validation*: After a coarse registration between the image and model has been made, we verify the presence in the image of the model objects. We proposed to use feature matching techniques for this step [1]. Model features that are not present in the image represent likely changes. Some missing features will be due simply to viewing conditions. These, however, can be predicted and explained from the site model itself. The task of finding objects in the image that are not in the model, is more difficult since the model is no longer as useful in directing the processing. We propose to do this by the next two steps.
- *Focus of Attention:* This will be a collection of techniques that will draw attention to significant structures in the image that are not explained by the existing model. To separate the significant changes from the insignificant ones, a *perceptual grouping* operation that organizes lower level features into higher level structures will be necessary [11,18]. Matching features between multiple images

(if available) will help distinguish between structures on the ground and above the ground. In monocular images, an important cue to significant changes will be the presence of shadows [1,9,14]. Collateral information may also be useful in determining the focus of attention.

- *Detailed Analysis:* In this step, we analyze in detail the structures indicated by the focus of attention processes. This step requires the development of automated or semi-automated site modeling techniques.

1.4 Classes of Objects

The following generic classes of objects are likely to be of interest:

- *Elongated Objects*: Objects such as roads, runways [8], rivers, railroads, powerlines, etc. Such features are typically characterized by large curvilinear features, though some, powerlines for example, may also require the ability to detect 3-D structures, such as the towers that support the powerlines as the lines themselves are not likely to be visible.
- *2-D Objects*: These correspond to large surface features such as water bodies, forest clearings and urban areas. Such objects are typically characterized by having uniform region properties such as intensity and texture.
- *3-D Objects:* The world is full of important 3-D objects, e.g. buildings, factories, bridges, powerline towers, etc. Such objects are perhaps the most important, however hardest to detect and describe automatically. Important clues to the presence of 3-D structures in monocular images are in the shadows and specific 2-D shapes for the contours of the objects. The latter clue is more easily applied if a specific model is available though generic models and can be used.
- *Mobile Objects*: Mobile objects are a special instance of 3-D objects. A complete 3-D model of such objects may not exist in the site model, hence transforming them to the new viewpoint, and comparing them with features in the image may not be possible. Also, these objects are rather small compared to structures such as buildings, therefore of limited resolution.

During the past year the first three steps of the process of change detection were worked on and applied to building structures. A technique was developed to precisely register the site model to a new image, and to verify the presence of model structures in the image (model validation). Pertinent details are given in Section 2. Separately, progress was made on automatic model construction without the use of the site model. This is required to detect building structures in the image that are not part of the model. Future work will bring these processes together in order to detect new structures and augment the site model accordingly.

This research has been directed, in part, by the imagery available for analysis, and by the importance of the problems as determined by experiments in Phase I of the RADIUS project.

2 Model Validation for Change Detection

The task of model validation in the context of change detection involves comparing a *new* image of a site (or a collection of images), to the information associated with that site. This information may consist of a site model and one or more previous images, and the results of previous analyses on these images. We assume that in all cases, a site model of suitable resolution and complexity is available.

In this Section we discuss progress on the problem of *model validation*, the process of confirming the presence of model objects in images. This task requires that we first register the new image to the site model, and then validate the model objects. We have chosen to work with features extracted from model information and the new image, rather than developing pixel based techniques.

Previous work on change detection relies on image differencing that is unable to separate the effects of changes resulting from different viewing conditions (such as different viewing positions, different illumination and seasonal changes) from important structural changes.

One example of previous work is by Lillestrand and Ulstad [13,20]. Given a reference image and a new image, the reference image is first corrected for any spatial distortion so that it is registered with the reference. This is done at each point by finding the best conjugate point on a simple correlation-based criterion. Global intensity corrections are made so that the first two moments of the image match, allowing for any differences in luminosity or film sensitivity. Next, a simple subtraction of the two images is made to reveal small scale changes between the two views. This algorithm works very well for this particular problem of detecting changes between two images.

Another approach is to use general classification algorithms after spatial registration to assign each pixel to a limited number of classes, thus transforming the intensity range of a pixel into a reduced number of levels. This assumes that preferably more than one band of image data is available, as in Landsat's multiple band images, for example. Depending on a pixel's position in the n-dimensional data space, and relying on the statistical description of each class (which is user-defined), a pixel is assigned a class. The classes could represent water, forest, roads, urban areas, and so on. Once the classification is done for each image, the detection of changes becomes straightforward.

Next, we give details of our model validation technique. Also discussed are experimental results applied to building structures using a site model and associated imagery supplied by the RADIUS program.

2.1 Validation of Buildings

Given a 3-D site model (consisting of a set of objects such as buildings, houses, and storage tanks, describing a site) and an image of this site, taken at a later time. The site model consists of geometric information only: we do not have any data about albedo or color for any of the objects. *Model validation* refers to the task of confirming the presence of model objects in the image. Specifically, we apply our technique to validating the *buildings* in the model. Extension to most objects of similar structure and geometry is straightforward.

Model validation requires three steps:

- *Coarse registration* of the model and the image. This is equivalent to finding the correct position and orientation of the camera at the time the image was taken.
- *Matching* model to image. Once the viewpoint is known, we project model features onto image coordinates, and use them for matching to equivalent features extracted from the image. We use the edges, or segments, of the wire-frame representation of the model to match against line segments extracted from the image. Matching features allow us to form hypotheses that represent the presence –or the absence– of a model feature in the image.
- *Validation* of hypotheses. We attempt to verify the hypotheses made with the help of the shadow information extracted from the image.

The following sections describe in more detail each of the three steps of our model validation algorithm.

2.2 Registration

The orientation and position (the external parameters) of the camera are usually known approximately for a given image. Otherwise, if the intrinsic parameters of the camera are known, –focal length and position of the principal point (the perpendicular projection of the focal point onto the image plane)– then it is possible to derive this information. This problem is known as *relative* orientation for the case of image-to-image registration, and *exterior* orientation for the image-to-model registration case.

2.2.1 Relative Orientation Between two Images

Most algorithms proposed for this problem in the literature are based on the use of a set of conjugate points (the matched control points) given by the user. In theory, given three parameters for rotation, three for translation, and allowing for an overall scale factor (which cannot be determined from two projected images), there are five constraints. The knowledge of a pair of conjugate points gives three equations (denoting the transformation of the three coordinates), but brings up two additional unknowns (the depth in both coordinate systems). Therefore, only five points are needed to compute the transformation (see [6]). In this case, however, the equations to be solved are non-linear.

These methods are usually computationally expensive, and in general, a unique solution is not possible when the minimal number of points are given. An example of a powerful non-linear solution can be found in [7].

It is possible, however, to solve the problem using only linear equations, if more information is given. The linear algorithms (Longuet-Higgins [16], reexamined later by Hartley [5] and Faugeras [4]), need at least 8 point correspondences. The main advantage of linear methods is that they are fast and guarantee the uniqueness of the solution, except in degenerate cases. These methods, however, exhibit sensitive behavior in the presence of real, noisy data.

2.2.2 Exterior Orientation Between an Image and a 3-D Model

The exterior orientation problem, which is the one we need to solve, is a special case of the relative orientation problem, thus, the algorithms mentioned above are applicable with only slight modifications. Instead of giving two sets of 2-D coordinates to describe the conjugate points, we will give, for a 'conjugate point', its 3-D coordinates in the site model, and its 2-D coordinates in the image.

Here we have more information than in the relative orientation problem. There are three rotation parameters and three translation parameters (in this case the overall scale factor can be determined). Each point brings up three equations as before, but only one additional unknown emerges. Therefore, in theory, it is sufficient to have only 3 points to solve the registration problem. However, the equations in this case, being non-linear, will result in more than one solution (up to eight); a fourth point is needed to completely solve the problem (see Horn, [6]). Using a larger number of points is desirable to improve accuracy of the solution with a least-squares method. We have found in our experiments that 20 reference points are adequate.

We have used two algorithms, both giving good results. First, we used the linear method of Hartley (see [5]) that we adapted to the exterior orientation problem. The non-linear algorithm we have used is the USGS resection algorithm provided as part of the RADIUS Common Development Environment (RCDE), the environment used for ARPA sponsored RADIUS research work, including our own Aerial Image Analysis research at USC. Both methods give a good coarse approximation of the orientation, with a preference for the non-linear method which gave results within a few pixels (see figure 2.7 on page 13 for an example of registration on image k4, using the USGS resection algorithm.)

2.3 Matching

In order to obtain an accurate registration between the model and the image, we first match features computed from the information in the site model with features extracted from the image. Secondly, each matching pair (model feature, image feature) denotes a hypothesis that the object the model feature is part of, has a corresponding object in the image. For this purpose we use the algorithm described in [17].

The algorithm assumes that there is only a translation between the two sets of features to match. We now briefly describe the matching process.

2.3.1 Input

We have chosen to match two sets of line segments. The first set consists of line segments extracted from the image using the LINEAR feature extraction system [19]. The second set is constructed from the 3-D wireframe model (edges of buildings and roofs) projected on the image, and consists of the visible segments only. Here, "visible" means segments of a model building that are not occluded by that same building (i.e. only self-occlusion is considered). Also, we consider segments longer than a certain length (10 pixels in our current system.)

2.3.2 The Matching Algorithm

We have a *candidate* pair of matching segments (one from the image set and one from the model set) when the two segments overlap, that is, the segment end points project inside the other segment. The segments also must lie within a certain "distance" of each other. The calculation of "distance" is as follows: If the two segments intersect, then the distance is zero. Otherwise the distance is the smallest projected distance of each segment end on the other segment, if it falls inside that segment. Figure 2.1 shows the distance between two segments p_1p_2 and p_3p_4. Each segment end point is projected onto the other segment. The distance is the minimum of the projection distances, if the projected point is inside the segment. In this case p_1 and p_4 project outside the other segment, therefore: $d(p_1p_2,p_3p_4) = min(d_2,d_3)$.

Each pair of candidate segments produces a *vote* as a function of the distance between the two segments and the differences of segment orientation and length.

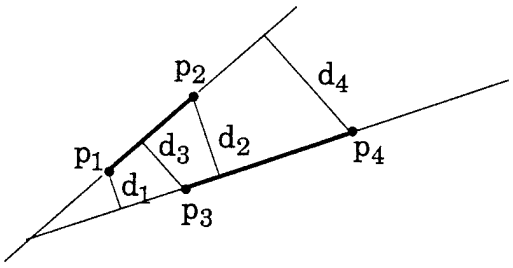


Figure 2.1 Distance between two segments p_1p_2 and p_3p_4.

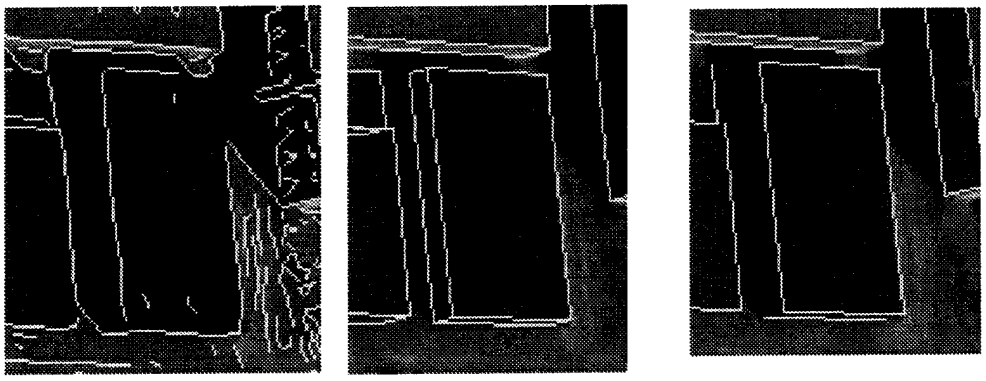
Votes are computed for candidate pairs and add their contribution into an accumulator array. The array axes denote translation. For noise effect reduction, the vote is cast not only at the point corresponding to the translation between the two segments (the translation between the two segment centers), but in a rectangular region around these points. For details see [17].

At the end of the voting process, a peak detector in the accumulator array gives the position of the best translation between the two sets of segments. Knowing the position of the peak, a second pass of the algorithm is used to collect the matching pairs

that contribute to the peak in the accumulator array. As a result of applying this two-pass process, we know for each segment of the model whether a corresponding segment was found in the image.

The next step in the validation process involves complete model objects. For each building object in the model, we have a record of the matching of its component edges. A *strong* match is denoted when at least four of the possible nine visible segments in its model have a corresponding match among the segments from the image. Otherwise the match is denoted as a *weak* match.

Objects having a strong match are considered validated. Figure 2.2 shows an example. The segments extracted from the image and the model are shown in figures Figure 2.2a and Figure 2.2b respectively. Figure 2.2c shows the segments of the projected model that were matched. This building is considered a strong match. Model buildings having weak matches, or no matches, require verification before they are considered validated. The verification technique uses shadows, detected and processed, using the techniques described in [14]. The next section gives details of the method.



(a) (b) (c)

Figure 2.2 Example of segment matching on building 36.

2.4 Verification of Objects

Objects having weak matches or no matches require additional verification in an attempt to confirm their presence in the image. Weak matches can be the result of several conditions, such as inaccurate registration, poor contrast or occlusion by other objects. Verification can be achieved by means of 3-D clues from stereo or from evidence of the shadows cast. Our current work processes monocular images and thus, uses shadow clues for verification.

We assume that the sun angles (direction of illumination and incidence angle) are known *a-priori*. Otherwise they can be computed from the time of the day at which the image was taken, and the latitude and longitude coordinates of the site. If the

time is unknown, the direction of illumination could easily be at least approximated by a trained operator from the image itself, after the registration has been completed. We also assume that the image contains reasonable shadow information.

To verify model objects we use the sun angles to generate the shadows they are supposed to cast. We then match these shadows to the shadow evidence found in the image. More details are given below. For this we make one assumption concerning the site itself. In the absence of a digital terrain elevation model (DTM), we consider the ground plane to be locally planar, so that the projections of shadows to the ground are simple to compute. Figure 2.3 shows the three shadow junctions and four boundaries cast by a "cubic" object. In the future we plan to use available DTMs for improved accuracy.

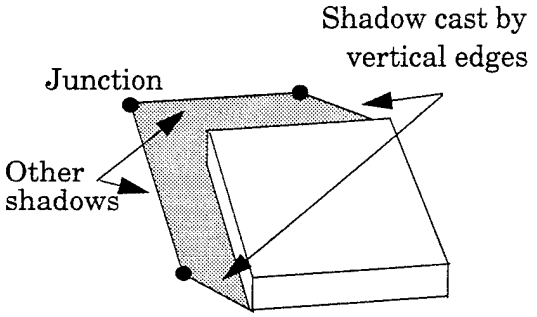


Figure 2.3 Typical shadows cast by an isolated "cubic" building.

Next we describe the verification process using shadows. It involves labeling image boundaries as potential shadows, and to compare these against shadows generated from model information.

2.4.1 Shadow Detection

We label image edges or segments as potential shadow segments by noting the consistency of the "dark" side of the segment with respect to the direction of illumination. Segments oriented parallel to the direction of illumination also correspond to possible shadow lines cast by vertical object edges. Similarly, we detect shadow junctions. The L-junctions formed (allowing for gaps) by potential shadow lines are labeled potential shadow junctions. For more details on the shadow labeling of segments and junctions see [10] and [11].

2.4.2 Validation of Hypotheses

The 3-D position of the objects having weak or no matches is known from the model. This information is used to predict their position in the image. Thus, the process consists of calculating the shadows cast in 3-D space and predict their location in the image. Then we search around the predicted locations for evidence of shadows among the potential shadows extracted from the image. If we find a sufficient evidence of shadows, then the presence of the building is confirmed.

Some objects may not be sufficiently isolated and shadows may not be cast on the ground. Also, shadows may be cast on other objects and sometimes will not be visible from the viewpoint. Figure 2.4 shows two examples of situations where building verification is made difficult by the presence of surrounding buildings. The shadow 1-3 cast by building C is not visible (occluded by building B), and the shadow 1-2 is projected on building A, not on the ground. We cannot take this into account until we have verified building B or C. Figure 2.4 shows building E occluded by building D, making verification difficult from this particular view point.

(a) (b)

Figure 2.4 Example of dificult building verification.

In our system we compare the evidence of shadows found with the visible shadow evidence. Visible shadows are determined by the knowledge of which objects have already been verified. In the "counting" (or accumulation) of shadow evidence, we also give greater importance to shadows cast by vertical edges and to shadow junctions that appear to correspond geometrically to shadow casting object structures. We consider this very strong and reliable evidence.

Not all weak matches are verified in a single pass due to poor contrast, occlusion, self-occlusion, lack of a DTM, and errors in the model itself. We, however, use the knowledge of previously verified buildings. This knowledge also allows us to predict more accurately the position of shadows on a subsequent pass. We repeat the process until no further verification of weakly matched objects is possible.

Note that for model buildings that have *no* matched segments, we are still able to predict their position in the image, by relation to the position of already confirmed buildings. We proceed in the same manner to verify these. In this case, however, the absence of shadows confirms the absence of the building. On the other hand, the presence of shadows does not guarantee the presence of a building. In this case attention is focused at this location for further study, including the application of object detectors, such as the one described in [11], to validate the presence of an object. This is a topic of current study, as part of the work on change detection.

2.5 Experimental Results

The validation system has been tested with a 3-D site model and images denoted as the Model Board 1 data set provided to us by the RADIUS program. The site model

describes geometrically a scene containing several buildings and other objects constructed at the approximate scale of 1:500. The images are 1320x1035 pixels with a ground sampling distance between 18.5 and 32.5 inches. Despite the 'artificiality' of this model (due to the indoor setting, no-clouds lighting, the use of small models for the buildings), there is sufficient added noise to consider the data set fairly realistic and adequate.

Figure 2.5 Model Board image k10.

A typical image is shown in Figure 2.5. The features in these images consist of buildings (of 'cubic' shape), houses (which are simple buildings with a gable) and storage tanks and chimneys (cylindrical shapes). The model contains 60 cubes and houses. Six cubes from the model were removed as they were considered too small or inaccurate. The basic shapes are shown in Figure 2.6, and a view of the complete site model is shown in Figure 2.7. We have used in our experiments the house and cube shapes, but extension to other similar shapes is straightforward.

2.6 Matching Results

The matching process gives excellent estimates of misregistration. We use an accumulator array of size 50 x 50 allowing a 25-pixel misregistration offset in any direction. Initial coarse registration using the USGS resection technique was in the order

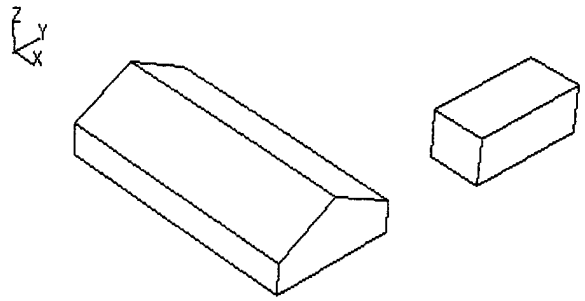


Figure 2.6 Two shapes used in our experiments with model board 1.

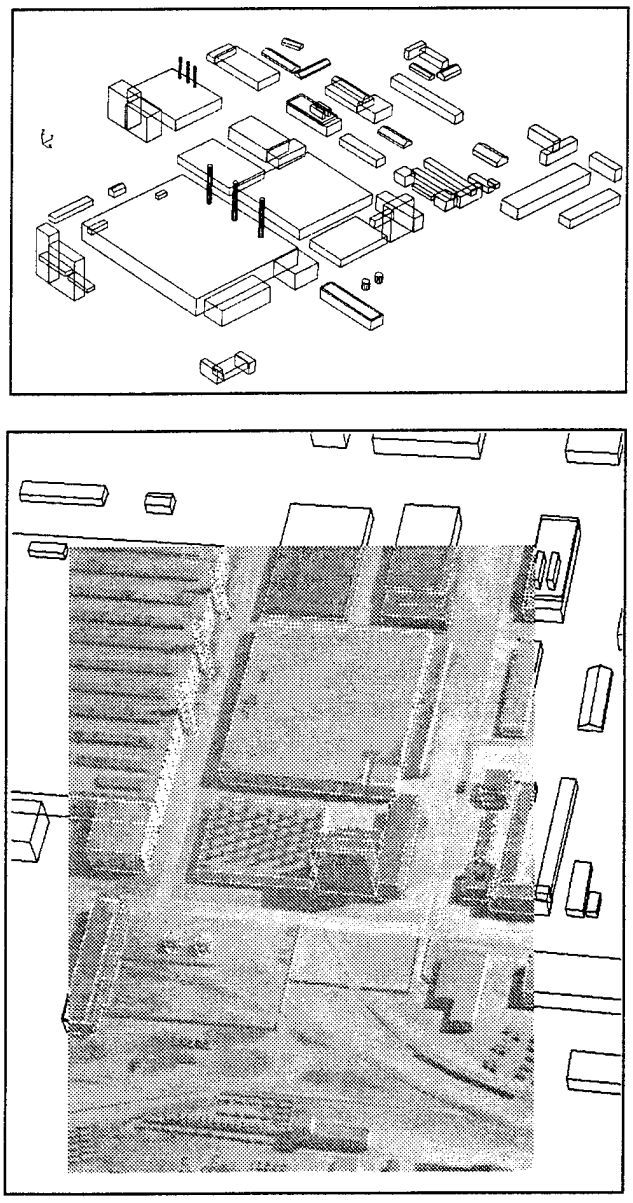


Figure 2.7 The model used in our tests, and the model registered to image k4.

of a few pixels. An example of accumulator array is shown in Figure 2.8. The peak is very sharp and allowing unambiguous determination of the translation. The two ridges in the horizontal and vertical dimension correspond to the dominant orientations of the segments in the model (the buildings are nearly all constructed parallel to each other.)

2.7 Summary of Verification Results

We summarize our validation results in Table 1. The model validation system was tested using 18 images (RADIUS data set j1-j8 and RADIUS data set k1-k10), at full and one-half resolution. As a global result, *all* the objects described in the model were present in the set of 18 images. The system software is written in Common Lisp and runs under the RCDE on a Sun SPARCstation 10.

Table 1: Summary of Verification Results

↓ Resolution / Stage →	Strong Matches (%) time	After Shadow verification (%) time
1/2 (~0.9 m/pixel)	47% (t1=7 min.)	80.1% (t2 = t1+5.5s)
1 (~0.45 m/pixel)	58.8% (t1=17.5 min.)	75.9% (t2=t1+6s)

The percentages are the averages after processing 18 images and show the percentage of validated buildings after the two stages of the algorithm: after the matching process (strong matches) and after the shadow verification process. Percentage of verified buildings for images j1-j8 and k1-k10. t1 is the time taken for edge detection and matching. t2 is t1 plus the time taken for the shadow verification process. This shows that t2-t1 is negligeable compared to t1+t2. Objects not visible in the image are easily determined and thus are not included in the figures. The images did not all have the same resolution, but we have grouped them into two groups of "approximate" resolution for convenience in the presentation of the results: 45 cm/pixel (full resolution) and, 90 cm/pixel (one-half resolution). The size of the images at full resolution is typically 1300x1000 pixels.

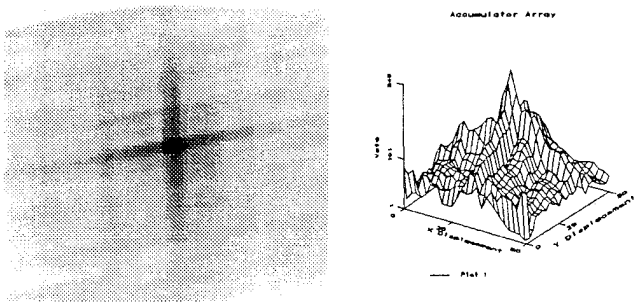


Figure 2.8 The accumulator array.

Note that at high resolution improved matching accuracy yields more strong matches as can be expected. Shadow verification however suffers as inaccuracies in the model are magnified. In some instances, the modeled height of an object is incorrect and thus, the projected shadows we derive fall long or short their actual locations in the image. This and other factors (the accuracy in registration, the quality and visibility of shadows, and the model itself) should be considered when evaluating the performance of the system.

Although our results are very encouraging, there is certainly room for improvement, both at the registration level and at the verification level. Inaccuracies in the model should be tolerated to a good extent at this stage, for example. These improvements will also help processing images that lack shadow definition or shadows altogether.

As far as processing time is concerned, we observe from the table that the time taken for shadow verification is negligeable compared to t1, which divides equally between the line extraction and the segment matching. We have to note that the times indicated here are given for comparison purposes only, no serious effort was made to optimize our implementation.

A typical result of our validation system is shown in Figure 2.9. The model objects are shown superimposed on the scene (J2 in this example). The objects shown in dashed lines are not visible in the image. The objects shown in bold lines were not validated. All others were validated. Note that non-validated objects correspond to dark buildings with poorly visible shadows.

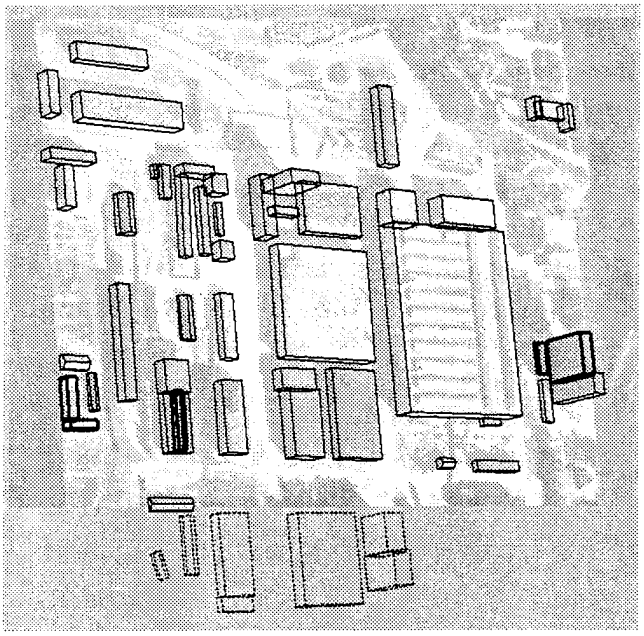


Figure 2.9 Validation result for image J2.

2.8 Conclusions and Future Work

We have presented a model-based system capable of verifying the presence in an image of objects described in a 3-D site model. This task, called model validation, is the first step towards the construction of a full *change detection* system.

We perform the model validation in three steps: first, register the image on the model, then perform segment matching and build hypothesis to represent the possible presence of each model object in the image, and finally, verify those hypothesis using the shadow information extracted from the image.

Although the results are very encouraging, there is certainly room for improvement, both at the registration level and at the verification level. This is part of our future plans. Inaccuracies in the model should be tolerated to a good extent at this stage, for example. These improvements will help processing images that lack shadow definition or shadows altogether.

Our plans for next year include additional testing of these techniques using imagery and models from different sites. We expect that site models for the modelboard 2 data set, Martin Marietta's Denver, Colorado site and Fort Hood, Texas' site be available. We also plan to incorporate the fast block interpolation projection (FBIP) method that allows testing with a variety of camera models, into the system. This will facilitate testing in operational environments where we expect to port our systems. The systems will be enhanced to include an appropriate user interface to allow other users to perform testing.

We also expect to be able to integrate the validation system and the model construction system to allow updating of the site model at the level of building structures.

Two other activities are planed for the coming year: limited testing of the validation system to verify the presence in the images of other objects, aircraft in particular, using simplified 3-D aircraft models derived by hand from a few views. The second activity involves the use of LOOM, a high-level reasoning tool developed by the University of Southern California (USC) Information Sciences Institute (ISI) under separate funding, to model context and events to be detected and described from a collection of images. The scenarios planned involve the deployment of mobile objects for specific purposes.

3 Detection of Buildings Using Perceptual Grouping and Shadows

The goal of this work is to detect and describe buildings from monocular views of arbitrary aerial scenes. This is a difficult but important task for many applications such as photo-interpretation, cartography and surveillance. Building detection is difficult for several reasons. The contrast between the roof of a building and surrounding structures, such as curbs, parking lots, and walkways, can be low. The contrast between the roofs of various wings, typically made of the same material, may be even lower. Low contrast alone is likely to cause low-level segmentation to be fragmented. In addition, small structures on the roof, and objects such as cars and trees adjacent to the building, will cause further fragmentation and give rise to "noise" boundaries. Roofs may also have markings on them caused by dirt or variations in material. Shadows and other surface markings on the roof cause similar problems.

Figure 3.1 A building from Ft. Hood, Texas

There are other characteristics of these images which may cause problems. Roofs have raised borders which sometimes cast shadows on the roof. This results in multiple close parallel edges along the roof boundaries. These edges often appear broken and disjointed. At roof corners and junctions of two roofs, multiple lines meet leading to a number of corners, thus making it difficult to choose a corner for tracking. A roof casts a shadow along its side and often there are objects on the ground such as grass, trees, trucks, pavement, etc., which lead to changes in the contrast along the roof sides.

Consider the building in Figure 3.1 (from a scene of Ft. Hood, Texas.) For simplicity, an overhead view is used as a running example. The building is easy for humans to see and describe, but it is difficult for computer vision systems. Figure 3.2 shows the line segments extracted from the image using LINEAR, our linear feature

extraction software [2, 19]. We are still able to see the roof structures of the buildings easily, but the complexity of the task now becomes more apparent. The building boundary is fragmented and there are gaps and missing segments. There are also many extraneous boundaries caused by other structures in the scene.

Figure 3.2 Line segments extracted from the image

There have been many previous attempts to solve this problem [9,10,11,12,15,18,21]. Building detection requires robust segmentation techniques and methods to infer the 3-D structure. These methods rely on edges or regions extracted from the image. Simple edge-based methods attempt to collect linked edge curves into the desired object boundaries, and succeed only for relatively simple scenes. Some edge-based methods have used some form of a contour tracing technique, see for example [9,10,21]. These are essentially local techniques that must make a decision of which path to trace at each local junction. Of course, all paths could be traced using backtracking, but the search space may become prohibitively large. Region based techniques construct closed curves that often do not correspond to the objects of interest.

Model based techniques can deal with fragmentation but require *a-priori* shape models. For example, it is not sufficient to say that the building is a rectangular parallelepiped; you must also supply the relative dimensions of the sides. In summary, these systems have shown interesting performance but on limited examples. None of these systems can generate a description of the buildings at the level of shape descriptions of the different wings.

We have proposed, instead, to use a perceptual grouping approach. Cultural features, such as buildings, represent structures that are not random but have specific geometric properties. In this we restrict the shapes of buildings to be a single or a composition of rectangular parallelepipeds (thus allowing L, T and I shapes for example).

Previous systems have assumed that the viewpoint is more or less overhead. The system described here uses the viewpoint angles (swing and tilt) needed to deal with images acquired from an oblique viewpoint. The geometric constraints relevant to shape take into consideration, as a function of the viewpoint angles, the expected

skewness of the rectangular surfaces that most buildings are expected to have. This property is used to organize the detected line segments into roof hypotheses. While the visible building sides (walls) can be hypothesized similarly, they are not handled now. This approach leads to fewer hypotheses than would be generated by a complete contour tracing scheme.

The approach combines several of the techniques from previous work. The perceptual grouping approach comes from the work described in [18], however, a very different hypotheses selection technique is used. Mohan and Nevatia, in fact, used perceptual grouping for stereo analysis --- here it is applyed to monocular analysis. The shadow analysis method is an extension of the approach first described in [9,10].

The diagram in Figure 3.3 shows the main components in the system. The system uses the line segments approximating the intensity boundaries to compute linear structures and relevant junctions among them. A hierarchy of features including parallel relationships and portions of skewed rectangles or parallelograms leads to the formation of building hypotheses. These consist of instances of rectangular shapes that potentially correspond to building roofs and parts of building roofs (see section 3.1). Next, promising parallelograms are selected and verified to correspond to roofs of building structures. Shadow information, if available, is used to help form, select and verify hypotheses. It also, as a function of the sun angles, provides estimates of the height of the structures, leading to a 3-D description of the scene.

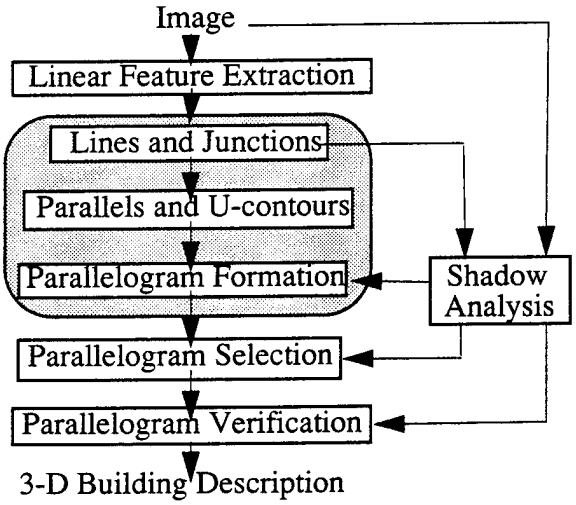


Figure 3.3 Block Diagram of the System

The philosophy in the design of this system has been to make only those decisions that can be made confidently at each level. Thus, we choose to generate as many hypotheses as seem feasible at the first level. The selection process, too, is conservative and favors keeping hypotheses that may be viable. The verification process has

the most global information and can make stronger decisions. Even here, if our system is to be embedded in a larger system, some of the decisions would be deferred to that system where more context is available for decision making.

The technique described in this report, we believe, significantly extends the range of scenes that can be analyzed, though many problems remain. We show several examples taken from the images provided by the RADIUS program to demonstrate the effectiveness of our technique. Also in the context of the RADIUS program we have transferred our software to two industrial sites and continue to support their testing tasks. The results, in general, have been very good, and the experience of technology transfer, albeit difficult , has been a successful one.

3.1 Generation of Hypotheses

The process of hypotheses formation is similar to the one described in [18] with the appropriate extensions to oblique views and the use of strong shadow cues (if available). In this process we construct a feature hierarchy which encodes the structural relationships specific to oblique views of rectangular shapes, presumably corresponding to the visible roof surfaces: Lines, skewed parallels, skewed U-contours, and skewed rectangles or parallelograms. The degree of skewness is computed as a function of the swing and tilt angles denoting the viewpoint. Figure 3.4 shows the angles involved. We expect that images from aerial scenes have a camera model associated with them from which these angles can be derived.

Next, we describe the hierarchy of features in the system:

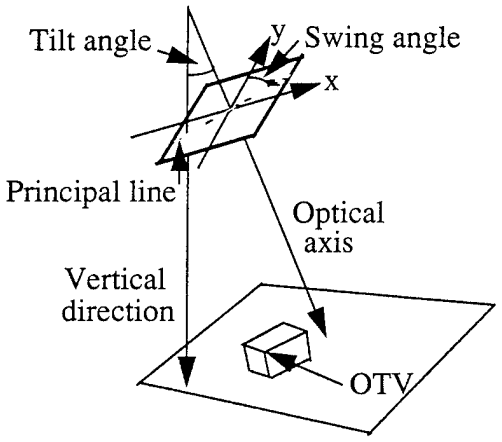


Figure 3.4 3-D Viewpoint angles

Lines and junctions

A group of close parallel lines represent a linear structure at a higher granularity level than the edges (see the common boundary between the building wings in Figure 3.2.) The resulting lines have a length and an orientation derived from the contributing elements. Figure 3.5 shows the lines obtained from grouping the segments in Figure 3.2. These lines are used to detect L-junctions and T-junctions also shown in Figure 3.5. For oblique views, we also look for evidence of vertical edges in the immediate neighborhood of the L and T-junctions, thus allowing us to detect potential OTV's. Vertical edges are detected by looking for line segments that are parallel to the image's principal line.

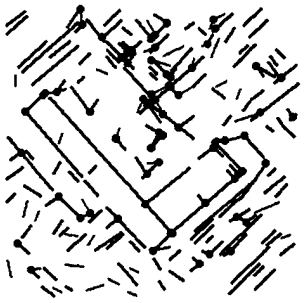


Figure 3.5 Linear structures and junctions

Parallels and skewed U-structures

Structures in urban scenes like buildings, roads and parking lots are often organized in regular grid-like patterns. These structures are composed of parallel sides. As a consequence, for each significant line-structure detected in the scene, there is not one, but many lines parallel to it. For each line, we find lines that are parallel and satisfy a number of reasonable constraints. Note that the formation of a parallel structure also aids in the formation of new lines, as they suggest extension and contraction of the parallels to achieve full skewed overlap.

When the two lines in a parallel structure have their ends aligned as a function of the viewing angles, they strongly suggest the presence of a line with which the parallel structure would form a skewed U-structure. Even if the third line does not exist in the set of lines, we hypothesize it and generate the U-structure.

Skewed rectangles or parallelograms

Skewed rectangle or parallelogram structures are generated from the U-structures. The parallelograms formed in our example are shown in Figure 3.6. In practical applications this number can be reduced by restricting the formation of parallelograms on the basis of size ---as a function of image resolution, for example. Parallelograms are also generated from matching junctions along the direction of illumination

(see strong junctions in section 3.3.) We hypothesize the missing portions of a parallelogram having a corner with a matching shadow corner or evidence of an OTV.

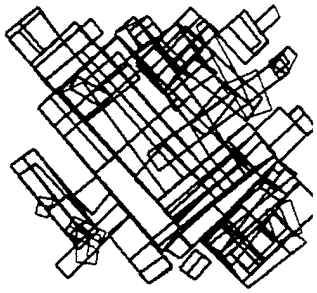


Figure 3.6 Parallelogram hypotheses generated

3.2 Selection of Hypotheses

After the formation of all reasonable parallelograms, a selection process is applied to choose parallelograms having strong evidence of support and having minimum conflict among them. Earlier versions of our system used a Constraint Satisfaction Network (CSN) [18]. In the current system, we use a criteria-based method which seems to give much more predictable results. Next we summarize our current method.

Our new system uses two kinds of criteria: local selection criteria and global selection criteria. Local selection criteria determine whether or not a parallelogram is "good" based on the local supporting evidence. Only good parallelograms are retained for global selection. It is possible that some of the good parallelograms retained after the local selection are mutually contained, duplicated or overlapped with some other good parallelograms. Global selection criteria select the best consistent parallelograms from good parallelograms.

We apply local selection criteria and global selection criteria differently. Local selection criteria (also called evaluation criteria) work together to evaluate the goodness of a parallelogram, while global selection criteria work separately. Each global selection criterion acts like a filter. The set of retained parallelograms pass through all filters and the set of parallelograms coming out from the last filter will be the set of parallelograms selected by the selection process.

The local selection criteria are used to remove parallelograms formed using weak evidence. For each parallelogram the evaluation criteria compute a goodness value. If this value exceeds a given threshold, the parallelogram is selected, otherwise the parallelogram is removed.

Every evaluation criterion is weighted according to its importance. The goodness of a parallelogram is then measured by the sum of the weighted values calculated by the evaluation criteria. The problem of measuring the goodness of a parallelogram

now becomes a problem of finding and formulating good evaluation criteria, and assigning appropriate weights.

Whether a parallelogram is good or not depends on the evidence of support. We distinguish between positive evidence, shown in Figure 3.7, and negative evidence, shown in Figure 3.8, of support for a parallelogram. The positive evidence of support includes the presence of edges, corners, parallels, OTV's and shadows.

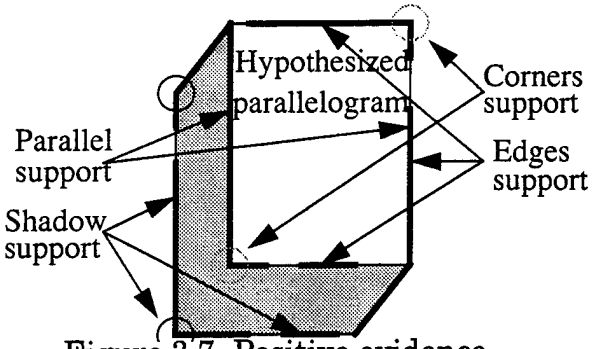


Figure 3.7 Positive evidence

The negative evidence of support includes the presence of lines crossing any side of a parallelogram, existence of L-junctions or T-junctions in any side of a parallelogram, existence of overlapping gaps on opposite sides of a parallelogram, and displacement between four sides of a parallelogram and its corresponding edge support. Negative evidence is as important as positive evidence because it helps us to remove those parallelograms which are less likely to be part of buildings.

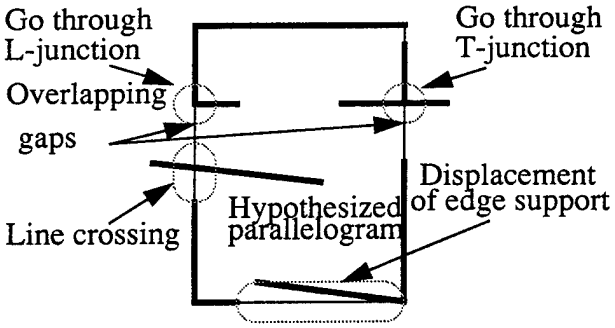


Figure 3.8 Negative evidence

Each kind of evidence of support is formulated into an evaluation criterion. There is no formal definition of goodness of a parallelogram, thus our evaluation criteria formulated from evidence of support are based on analysis of likely and unlikely events. For example, four junctions are very unlikely to fall on the four corners of a parallelogram accidentally. So the existence of four corners on a parallelogram strongly suggests that the parallelogram is good. Also, from the Gestalt Laws of Perceptual Grouping, the Law of Closure suggests that the existence of L-junctions or T-junctions on a side of a parallelogram will make a closure on part of the parallelogram, meaning

that the hypothesized parallelogram is not good. Some evidence of support is not always available, such as the shadow evidence and the OTV corner evidence, but they are important because it is very unlikely that some shadow features will appear around the hypothesized parallelogram, by chance, and the probability for three lines to form an OTV corner, by chance, is very small. We can emphasize the importance of an evaluation criterion by assigning a higher weight to it.

Positive weights are assigned to those evaluation criteria formulated from positive evidence of support, while negative weights are assigned to those evaluation criteria formulated from negative evidence of support. A weight should be assigned to each evaluation criterion according to the probability of existence of buildings under the condition of presence of the evidence of support from which the evaluation criterion is derived. However, we do not have the probabilistic analysis of goodness of a parallelogram, but the problem of optimal weights assignment for a given set of examples could be formulated into a search problem.

Good parallelograms surviving local selection may compete with each other. For example, some parallelograms could share the same edges or corners support and some parallelograms might overlap with each other. The goal of global selection criteria is to select a minimum set of parallelograms which best describe the rectangular composition of the scene.

Global selection criteria examine overlapping parallelograms and choose one if appropriate. The selection is based on relative properties of each parallelogram, the amount and kind of overlap, and whether they share support or not. Note that a parallelogram fully contained in another is not necessarily removed. If a parallelogram does not overlap with any other parallelogram, then it is not in competition, and it remains. There are four global selection criteria in our system. They are the criterion for duplicated parallelograms, the criterion for mutually contained parallelograms, the criterion for fully contained parallelograms, and the criterion for overlapping parallelograms.

It is very easy to extend and improve the criteria-based selection process. If a new kind of evidence of support is found to be crucial for the goodness of a parallelogram, we can formulate an evaluation criterion from the evidence of support and merge the evaluation criterion to the original set of evaluation criteria by assigning appropriate weight to it and adding the weighted value to the goodness value. On the other hand, if a new global relationship between parallelograms is found to be important, we can also implement a new filter to enforce the relationship and add the new filter in appropriate position to the original pipeline of filters.

The parallelograms selected in our Ft. Hood example, after both the local and global selection criteria have been applied, are shown in Figure 3.9.

3.3 Verification of Hypotheses

The purpose of verification is to validate the selected hypotheses to correspond to buildings. Our validation step segments the objects, generates a description of the

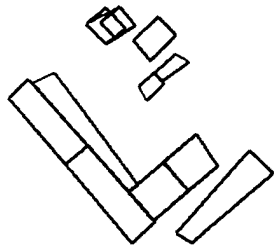


Figure 3.9 Selected parallelograms

shape of the structures and derives a 3-D model. The use of shadow evidence, discussed below, uses methods described in [9,10 ,11] with the appropriate extensions to handle oblique views. Oblique views require at least two sun angles (see Figure 3.10), the direction of illumination and the sun incidence angle. For testing, we have gotten these angles from image measurements.

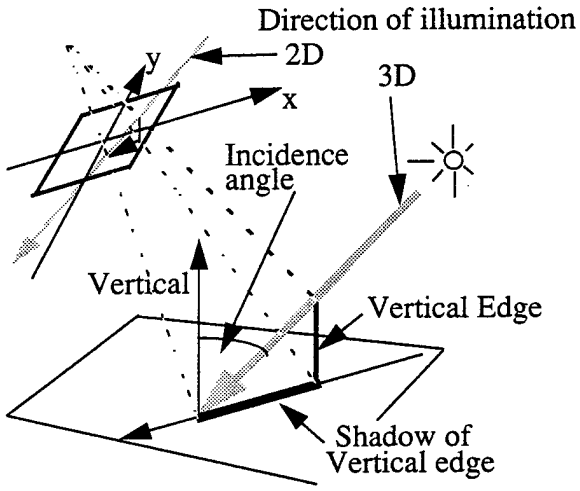


Figure 3.10 Sun angles & oblique shadow geometry

3.4 Shadow Analysis

Shadow analysis is the establishment of correspondences between shadow casting elements and shadows cast, and the use of these correspondences to verify and model 3-D structures. We assume that the ground surface in the immediate neighborhood of the structure is fairly flat and level. The shadow casting elements are given by the sides and junctions of the selected parallelogram hypotheses. The shadow boundaries are located among the lines and junctions computed earlier from the image.

There are a number of difficulties that prevent the accurate establishment of correspondences, however. Building sides are usually surrounded by a variety of objects, such as loading ramps and docks, grass areas and sidewalks, trees, plants and shrubs, vehicles, and light and dark areas of various materials. Nearby structures

may reflect light into the shadowed areas making the objects in it more visible, and so on. To deal with these problems we have adopted the following definitions, criteria and geometric constraints to analyze the shadows adjacent to parallelograms (see Figure 3.11):

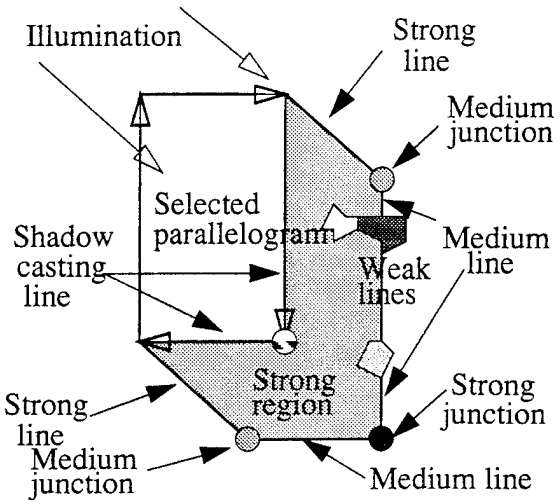


Figure 3.11 Shadow features

Strong Junctions: Matching junctions along the direction of illumination, having a consistent shape and a consistent attitude. These junctions constitute the strongest monocular cue to the presence of a 3-D structure. We use knowledge of these correspondences also to help form and select parallelogram hypotheses.

Strong Lines: Shadow boundaries cast by vertical object edges. We use this evidence also during hypotheses formation and selection.

Medium Lines: The parallelogram sides that are supposed to cast shadows must have corresponding shadow lines.

Medium Junctions: The junctions formed by strong and medium lines, found along the direction of the strong lines.

Weak Junctions and Lines: Junctions and breaks in the shadow boundaries between the strong and weak junctions.

Strong Regions: Dark regions surrounded by strong and medium junctions. We require that this region be darker than the parallelogram region regardless of their gray level.

Weak Regions: In the absence of geometric correspondences of junctions and lines, a dark region adjacent to parallelogram, consistent with the direction of illumination.

3.5 Shadow Process

The shadow process consists of four steps:

Extraction of Potential Shadow Evidence

Potential shadow evidence consists of lines, junctions and intensity statistics. We extract the following:

- Lines parallel to shadow boundaries cast by vertical edges. They represent potential shadow lines cast by 3-D structures in the image.
- Lines having their dark side on the side of the illumination source are potential shadow lines.
- Junctions among the lines above.
- Pixel statistics to compare relative brightness.

The potential shadow lines and junctions extracted from the lines in our Ft. Hood example are shown solid in Figure 3.12. The underlying edges are shown in gray.

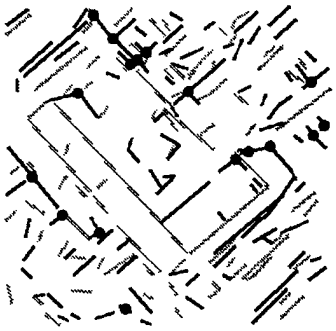


Figure 3.12 Potential shadow lines and junctions

Search for Shadow Evidence

For each parallelogram we look in a search window (dashed lines in Figure 3.13) and collect all the potential shadow evidence in it. The search distance is arbitrarily chosen as a function of the maximum expected building height and the sun incidence angle. There is the possibility that lines, not relevant to the current parallelogram, be included. They however, have a reduced effect in the presence of the real evidence.

Medium and weak lines that are parallel to the parallelogram side are favored. In some cases there may be various sets of lines, all parallel to the building side, but at various distances from the parallelogram side. This is actually a common occurrence since many sidewalks, grass areas, streets, vehicles and so on, will be found to be arranged or located parallel to building sides. In this case we choose those shadow

lines at the distance from the parallelogram side such that the sum of their lengths is greater, but not exceeding the length of the parallelogram. We determine the width of the shadow by averaging the distance to the lines selected. The selected evidence is then considered to surround the shadow region. We compute the mean intensity of this region and compare it to the parallelogram region.The evidence collected for both sides is combined to give the evidence for the parallelogram.

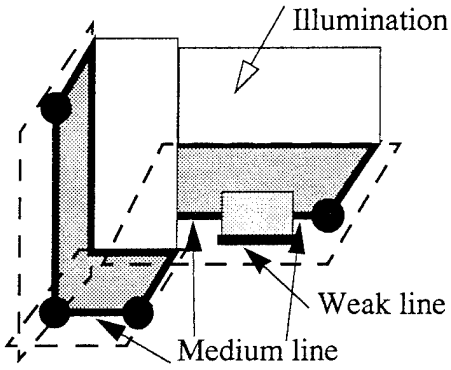


Figure 3.13 Windows to search for shadows evidence

Evaluation of Shadow Evidence

We evaluate the shadow evidence and give a confidence value as a weighted sum of the evidence of strong junctions, medium junctions, strong lines, weak lines, and strong and weak regions. We designated five levels of confidence. Each level of confidence requires that a minimum amount of the different kinds of evidence be present. Very high confidence requires that every kind of evidence be detected. Very low evidence is reported when no geometric correspondences can be established but the presence of a region, adjacent to and darker than the parallelogram region itself, is found. The parallelograms selected on the basis of shadow evidence are shown in Figure 3.14.

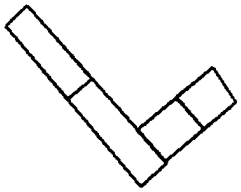


Figure 3.14 Verified parallelograms

Use of Shadow Evidence

The parallelograms verified by shadows are used to generate an image containing the corresponding regions. The pixel values inside these regions encode the esti-

mated height (as a function of the estimated shadow width and the sun incidence angle), thus giving an "elevation map" of the scene. This image can be viewed from an arbitrary viewpoint. The transform that projects the 3-D scene onto the 2-D screen for viewing can then be used to collect the pixel values from the input image, and use them to "paint" or render the various regions in the elevation map. Other 3-D representations, such as wire frame models, can also be easily derived from the knowledge associated with the detected and verified parallelograms. A 3-D rendered arbitrary view computed from the parallelograms verified in the Ft. Hood example is shown in Figure 3.15.

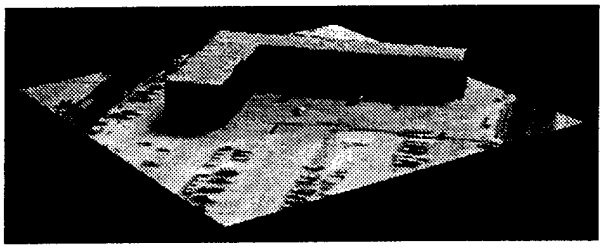


Figure 3.15 3-D view from another viewpoint

3.6 Results

The system has been tested on a number of examples provided by the RADIUS program with good results. A few are shown to demonstrate the performance of the system and point out some of the sources of problems. As part of the RADIUS program, the system has also been ported to run on UNIX workstations and transferred to two industrial sites and tested on some operational imagery. The results have been very promising and potentialy useful to the intended users. The speed of processing is a limitation however. It takes from 2 to 5 minutes to process a 512x512 image containing a few buildings. A 1320x1100 image with about 40 structures takes about one hour on a Sparc10/30. USC has a group currently working on parallel implementation of vision algorithms such as our system.

In the following figures, (a) shows an image, (b) the line segments extracted from it, (c) the linear structures and junctions computed, (d) the parallelograms hypothesized, (e) the selected hypotheses, and (f) the hypotheses verified by shadows. In particular, note figure (e), the excellent performance of the new selection technique. In the absence of shadow information, the selected parallelograms can be matched by our system if stereo views are available, thus, providing verification and a 3-D model.

Figure 3.16 shows a set of four buildings and part of another. The difficulty is with the building with the patterned arrangement of small objects on the roof. The shadows cast by these reach one side of the building causing it to be fragmented. The shadow occluding the top left corner of the building and the poor boundary definition on the top right are also a source of difficulty. Accurate junction information can not

be established and the systems must hypothesize a portion of the building. The strong shadow cues, however, help form parallelogram hypotheses for most of the building.

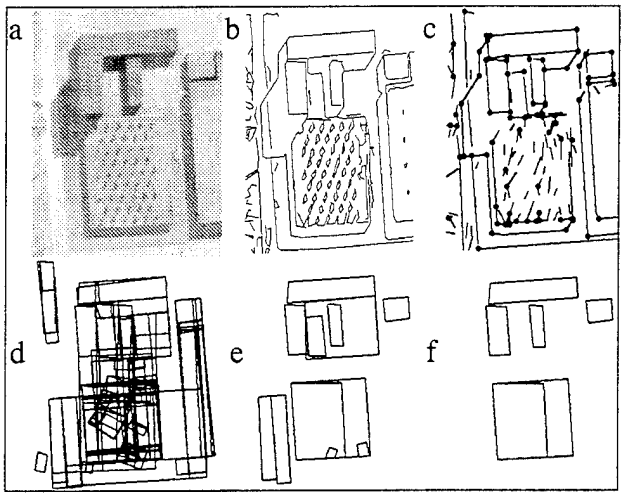


Figure 3.16 Modelboard - Scene 1

Figure 3.17 shows an oblique view of two dark buildings. The boundaries between dark buildings and shadows usually have low contrast and are difficult to detect.

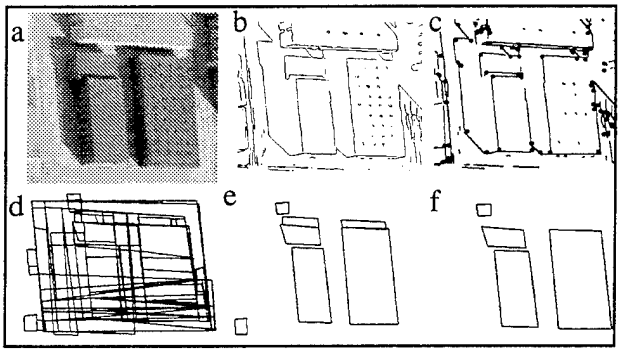


Figure 3.17 Modelboard - Scene 2 (oblique)

Figure 3.18 shows a complex building with numerous rectangular components on the roof. We are able to exploit the presence of strong shadow evidence here. It allows the system to form a hypothesis for the entire building in spite of the broken and fragmented boundaries. Note that the selection mechanism is able to select most of the rectangular components on the roof as well. Figure 3.18g shows a 3-D rendered view of the building, from an arbitrary viewpoint.

Figure 3.19 shows another oblique view including some simple buildings. Note that the considerable fragmentation of the roof boundaries due to the features, such as windows, on the visible sides is tolerated well and reconstructed properly by the colinearization grouping.

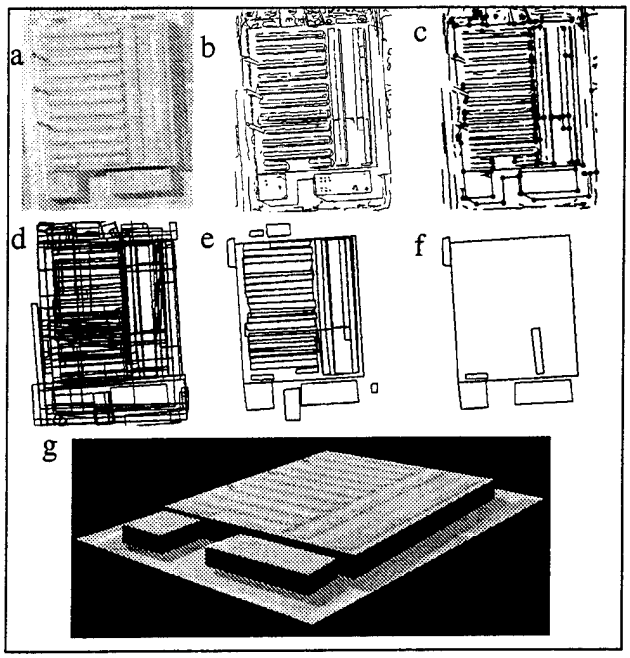


Figure 3.18 Modelboard - Scene 3

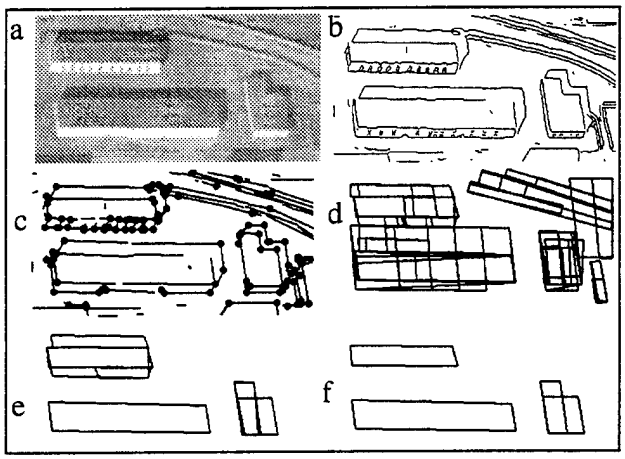


Figure 3.19 Modelboard - Scene 4 (oblique)

Figure 3.20 shows a building in Ft. Hood where some of the details of one of its sides is visible, apparently doors. These and the vehicles parked on the other side result in highly fragmented boundaries. The parallelograms verified by shadows include one that is formed from various aligned parked trailers which collectively cast a shadow. The small parallelogram on the bottom has a strong shadow junction corresponding to an actual narrow shadow cast by a vehicle. The lower wing of the building has a strong line and a corresponding medium junction. The rest of the shadow is diffused and is visible as a "dark" region adjacent to the building wings with no definite boundaries.

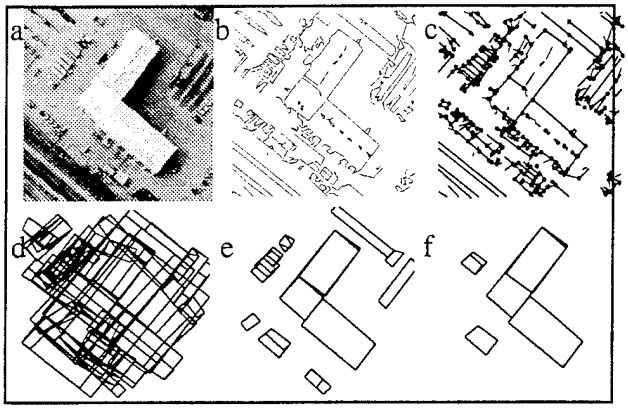


Figure 3.20 Fort Hood - Scene 2

In Figure 3.21 the I-shaped building has no strong evidence of shadows. The parallelograms are weakly validated on the basis of a strong region (shadow) which up to a given maximum search distance remains "strongly" dark.

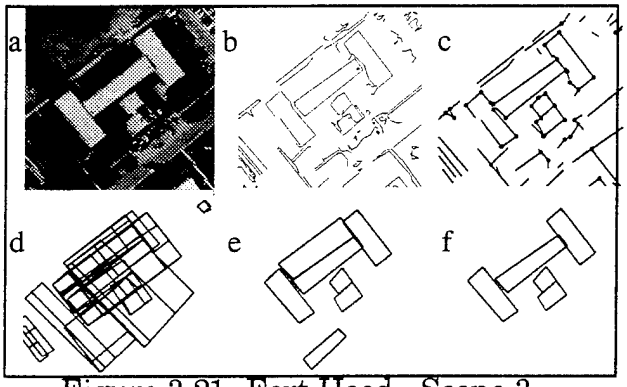


Figure 3.21 Fort Hood - Scene 3

Figure 3.22 shows a group of small buildings arranged in a parallel fashion, and surrounded by other parallel structures. In spite the large number of hypotheses the system is able to select the relevant ones.

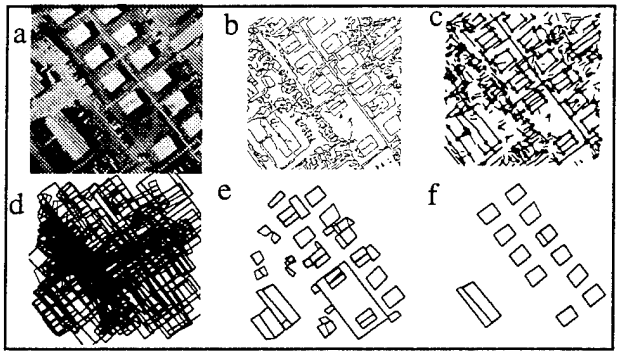
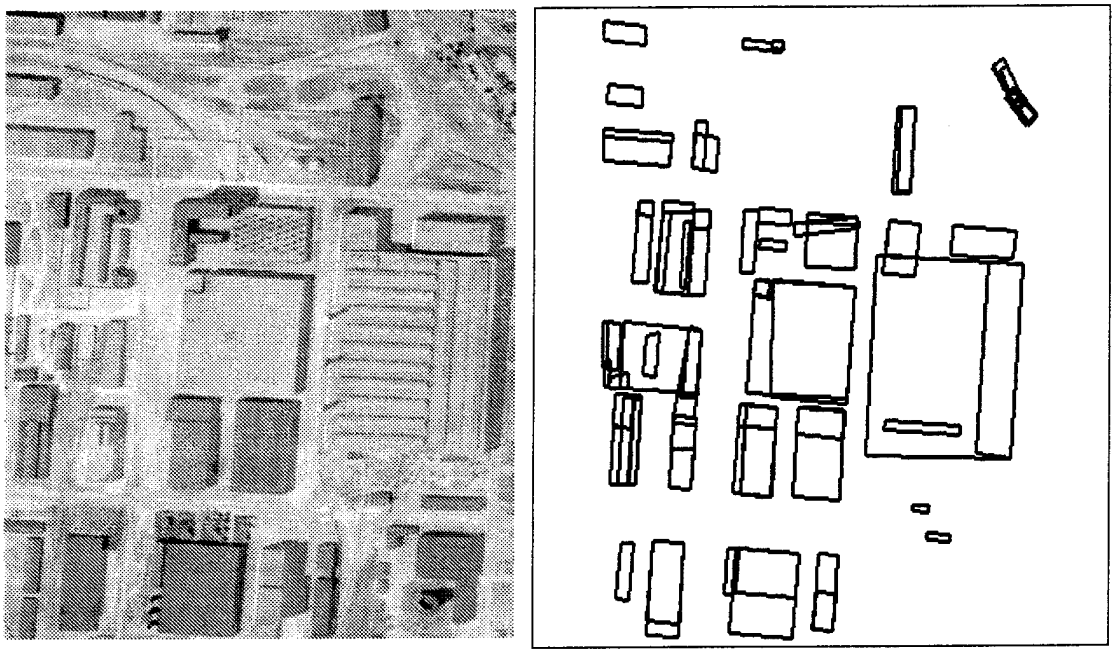


Figure 3.22 Fort Hood - Scene 4

Finally, Figure 3.23a shows an image from the RADIUS modelboard set containing a large number of structures (about 40). The system forms 1,724 hypotheses and selects 177. Some rectangles are selected but not verified. These correspond to dark low buildings with a small shadow that becomes merged with the buildings roof, and thus, becomes harder to verify. The system verifies 112 hypotheses on the basis of shadow evidence. Those verified on weak evidence (no object-to-shadow correspondences were possible) are excluded from the set of 75 shown in Figure 3.23b. We have not implemented a step that combines these rectangles into structures yet. The rectangles verified, however, represent a large majority of the components of the 40 or so structures in the image. Portions of the dark building on the lower right part of the image were only weakly hypothesized, and thus, not selected for verification.



(a) (b)

Figure 3.23 Modelboard image and verified buildings

3.7 Conclusion and Future Work

We plan to continue to extend our current system to detect the visible sides of buildings from oblique views of the scenes. This requires additional work in the use of the OTVs that can be located. This will allow us to rely less on the shadow evidence as it becomes more difficult to establish object-to-shadow correspondences. With oblique views, the shadows are likely to be occluded by the objects themselves or fall onto regions that belong to nearby structures. Currently, we assume that the detected and verified structures lay on the ground. Some structures, however, are located on top of other structures. That level of refinement of the description requires an additional step in our system and is one of the subjects of our current and future work.

4 References

[1] M. Bejanin, A. Huertas, G. Medioni and R. Nevatia, "Model Validation for Change Detection," Proceedings of the 1994 ARPA Image Understanding Workshop, Monterey, California, November 1994, pp. 287-294.

[2] J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), November 1986, pp. 679-698.

[3] C. Chung and R. Nevatia, "Recovering Building Structures from Stereo," In Proceedings of the IEEE Workshop on Applications of Computer Vision, Palm Springs, California, December 1992, pp. 64-73.

[4] O. D. Faugeras, Q.-T. Luong, S.J. Maybank. "Camera self-calibration: theory and experiments," Proceedings of the 2nd European Conference on Computer Vision, Santa Margharita, Ligure, Italy, June 1992, pp. 321-334.

[5] R. I. Hartley, R. Gupta, T. Chang. "Stereo from Uncalibrated Cameras," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Champaign, Illinois, June 1992, pp 761-764.

[6] B. K. P. Horn. "Robot Vision," Mc Graw Hill, 1986.

[7] B. K. P. Horn. "Relative Orientation Revisited," in Journal of the Optical Society of America A., Vol. 8, Number 10, October 1991, pp. 1630-1638.

[8] A. Huertas, W. Cole, and R. Nevatia, "Detecting Runways in Complex Airport Scenes," Computer Vision, Graphics, and Image Processing, Vol. 51, No. 2, August 1990, pp. 107-145.

[9] A. Huertas, "Using Shadows in the Interpretation of Aerial Images," University of Southern California USC-ISG Technical Report 104, October 1983.

[10] A. Huertas and R. Nevatia, "Detecting Buildings in Aerial Images," Computer Vision, Graphics and Image Processing, 41(2), February 1988, pp. 131-152.

[11] A. Huertas, C. Lin, and R. Nevatia, "Detection of Buildings from Monocular Views of Aerial Scenes using Perceptual Organization and Shadows," Proceedings of the 1993 ARPA Image Understanding Workshop, Washington D.C., April 1993, pp. 253-260.

[12] R. Irving and D. McKeown, "Methods for exploiting the Relationship Between Buildings and their Shadows in Aerial Imagery," IEEE Transactions on Systems, Man and Cybernetics, SMC 19(6), November/December 1989, pp. 1564-1575.

[13] R. Lillestrand, "Techniques for Change Detection," IEEE Transactions on Computers, Vol. C-21, Number 7, July 1972, pp. 654-659.

[14] C. Lin, A. Huertas, R. Nevatia. "Detection of Buildings using Perceptual Grouping and Shadows," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, Washington, June 1994, pp. 62-69.

[15] Y. Liow and T. Pavlidis, "Use of Shadows for Extracting Buildings in Aerial Images," Computer Vision, Graphics and Image Processing, 49, pp. 242-277.

[16] H.C. Longuet Higgins. "A Computer Algorithm for Reconstructing a Scene from Two Projections," Nature, Vol. 293, September 1981, pp. 133-135.

[17] G. Médioni, A. Huertas, M. Wilson. "Automatic Registration of Color Separation Films," Machine Vision and Applications, Springer-Verlag, New York, Vol. 4, 1991, pp. 33-51.

[18] R. Mohan and R. Nevatia, "Using Perceptual Organization to Extract 3-D Structures," IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(11), November 1989, pp. 1121-1139.

[19] R. Nevatia and K. Babu, "Linear Feature Extraction and Description," Computer Graphics and Image Processing, Vol. 13, 1980, pp. 257-269.

[20] M.S. Ulstad. "An Algorithm for Estimating Small Scale Differences Between Two Digital Images," Pattern Recognition, Vol. 5, 1973, pp. 323-333.

[21] V. Venkateswar and R. Chellappa, "A Framework for Interpretation of Aerial Images," Proceedings of the International Conference on Pattern Recognition, Atlantic City, New Jersey, June 1990, pp. 204-206.